Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 31 (2014) 711 – 720

Procedia Computer Science

Information Technology and Quantitative Management (ITQM 2014)

Sinkhole attack detection based on redundancy mechanism in wireless sensor networks

Fang-Jiao Zhang[a,b], Li-Dong Zhai[a,*], Jin-Cui Yang[b], Xiang Cui[c]

[a] National Engineering Laboratory for Information Security Technologies, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

[b] Beijing University of Posts and Telecommunications, Beijing 100876, China

[c] Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China

Abstract

Wireless sensor networks has a bright future because of its low-cost, save-power, and easy implementation .etc. However, its security problems have become hot research topics in many applications. Sinkhole attack is just one of frequently encountered security problems, which is easily combined with other attacks to cause more damage. In order to prevent sinkhole attack, we do some research on it, and one way to detect the sinkhole attack based on the redundancy mechanism is proposed in this paper. For the suspicious nodes, messages are sent to them through multi-paths. By evaluating the replied comprehensively, the attacked nodes are finally confirmed. Lastly, a simulation is performed to test the effectiveness of the method. And the simulation shows that the approach could work to some extent.

Selection and peer-review under responsibility of the Organizing Committee of ITQM 2014.

Keywords: Sinkhole attack; detection; redundancy; wireless sensor networks

1. Introduction

Wireless sensor networks are a multi-hop temporary autonomous system made up of a group of mobile nodes with wireless transmitters and receivers. Not relying on any preset infrastructure, it would achieve automatic organization and running in arbitrary mesh topology. Together with micro-processing and wireless communication capabilities, they are widely used on occasions which require rapid deployment and dynamic networking, such as military tactical communications and emergency communications. They are becoming a research subject of critical significance in practical application.

* Corresponding author. Tel.: +0-000-000-0000 +086-010-82546733; fax: +086-010-82546701.
E-mail address: zhailidong@iie.ac.cn.

Selection and peer-review under responsibility of the Organizing Committee of ITQM 2014.
doi:10.1016/j.procs.2014.05.319

However, wireless sensor networks are vulnerable to various types of attacks, including Sybil, selective forwarding, Sinkhole、Wormhole and HELLO FLOOD. Sinkhole attack, mainly discussed in this paper, is a relatively common attack. The nodes attacked claim to be able to provide a single-hop, high-quality path to the base station, which attracts the neighbor nodes to change the original route. And packets sent to the base station are discarded or forwarded to the sinkhole attacker, thus seriously damaging the load balancing of the network. It is easily combined with other attacks, causing greater damage to the network. There are some defects in the former Sinkhole detections [1-8], for example, the cooperative malicious nodes could not be detected and the detection algorithm would be complicated. Thus here we raise a new one based on redundancy mechanism.

The rest of the paper is organized as follows. In Section 2, some research work before about sinkhole attack some research work about multi-path selection- the core of the algorithm is introduced briefly. In Section 3, the new approach we raise is discussed thoroughly, a key point in the paper. Section 4 is the simulation of the algorithm. Finally, we summarize our work.

2. Detection Algorithm

Path in the typical multi-path selection in wireless sensor networks is the connection between nodes. And either of them is the Sink node. The process of the path establishment consists of three stages: route request, route reply and route establishment [9, 10]. Here, the Sink node is node A or node B.

(1) route request

Node A in the network broadcasts route request packet to node B. All nodes in the communication range node of A would receive it. The packet contains the field named Path to record the path information, a collection of nodes the packet passes. In this process, nodes would record neighbor nodes and update their neighbor lists until the route request packet reaches node B.

(2) route reply

When the node firstly receives the route request packet, it would save the sending node as a parent node, and add their own identity to the field-Path in route request packet, and then transmit the packet. In this stage, node B receives the packet and would send the route reply packet to its parent node including the neighbor nodes. Likewise, other nodes would also do it until the node is node A.

(3) route establishment

Node A would build the network topology according to the received packet information. Then multi-path between node A and node B are calculated according to Dijkstra shortest path algorithm.

It's supposed that nodes in the wireless sensor networks are static and we have known suspicious nodes in the wireless sensor networks. Before the algorithm, some definition are given.

Definition 1. Set S={1, 2, 3, 4……N}, which means all of the nodes in wireless sensor networks.

Definition 2. Set S_{sus}={1, 2, 3,……m}, which means the suspicious nodes in the networks.

Definition 3. Set S_{mal}={null}, which means the malicious nodes in the networks.

Apparently, there is a relation between sets: $\mathbf{S \supseteq S_{sus} \supseteq S_{mal}}$.

The task is to identify which nodes are the malicious nodes. The Sinkhole detection algorithm proposed here based on redundancy mechanisms is to build M disjoint paths between the original node and the suspicious node.

2.1. Original nodes selection

(1) Source nodes selection

We learn from the existing multi-path establishment methods and give our own algorithm. Firstly, we should know how to select the original nodes.

Definition 4. Set S_{cre}={null}, which means the credible nodes in the networks. And $\mathbf{S_{cre}=S-S_{sus}}$.

Definition 5. Set $\mathbf{S_{ori}}$={null}, which means the original nodes we will select in the networks. And it includes M nodes randomly chosen in the set $\mathbf{S_{cre}}$, that is $\mathbf{S_{ori}}$=C(M, $\mathbf{S_{cre}}$).

Definition 6. Set $\mathbf{S_x}$={null}, which would be used later.

(2) Multi-path selection

Because the network is static, we can know the position of all the nodes in it. We imagine the Sink node would record the coordinate information of all the nodes as shown in Table 3-1.

Table 1. Coordinate information of nodes

Node ID	X	Y
1	x_1	y_1
2	x_2	Y_2
.....	.....	.....
N	x_n	y_n

In the Table 1, N signifies the number of nodes in the network. We note the position of Sink node is (0,0) and other nodes' position would be calculated according to it. And the position of node i (i ∈[1, N) is(x_i, y_i) .

Considering processing power, storage and communication capacity of the Sink node, we assign more tasks to Sink node and make Sink node calculate the M paths between nodes. We also use the typical three phases - routing request, routing feedback and routing establishment to describe the process of paths building. And the paths is between node A in set $\mathbf{S_{cre}}$ and node B in set $\mathbf{S_{sus}}$.

- routing request

Node A only need to send a "routing request packet" to the Sink node. The reason why quotation marks is that it is different from the routing request packet mentioned in Section 2. The packet is forwarded to the Sink node instead of Node B. The main role of the packet is to inform Sink node to calculate the M shortest path between node A and node B.

- routing reply

The main idea of the algorithm is to calculate shortest paths between node A and node B by Sink node. It's obviously that the shortest distance between two points is a straight line. So we can easily know that the shortest corresponding paths are composed of the nodes near to the straight line. The process is divided to two steps:

Step 1. Straight line calculation

According to Table 1, node A is known in(x_A, y_A) and the node B is known(x_B, y_B). The straight line between them is expressed as the formula (1):

$$y = \frac{y_B - y_A}{x_B - x_A} * (x - x_A) + y_A \tag{1}$$

Straight line is represented by the blue line in Fig 1. And the scope of the nodes looking for on the path between node A and node B is the rectangle as shown in Fig 1.

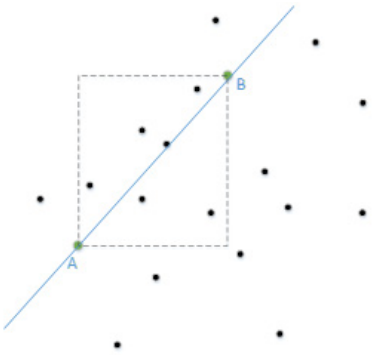

Fig. 1. Effective area is calculated path

Specific coordinates satisfy the following equation:

$$\begin{cases} X_A < X_i < X_B \\ y_A < y_i < y_B \end{cases} \tag{2}$$

Step 2. Path nodes selection

Obviously, we can select the nodes within the rectangular area shown in Figure 3-1. The nodes are ordered by position X of the node ascending. And for next node, we choose the node which is the communication range of the nodes x_i and whose abscissa is as big as possible and vertical axis is nearest to the straight line. A satisfying node can match the following rule:

$$\left| y_i - \left(\frac{y_2 - y_1}{X_2 - X_1}\right) * (X - X_i) + y_1 \right| < \left| y_j - \left(\frac{y_2 - y_1}{X_2 - X_1}\right) * (X - X_j) + y_1 \right| \tag{3}$$

When $xi \simeq xj$, if and only if formula（3）is true，the node is node i; otherwise, the node is node j.

The chosen nodes would be selected one by one and they would be saved in an array path[N]. The above is sorted by abscissa x-axis; alternatively, you can also choose the y-axis. Besides, the paths calculated by the two ways are compared. And the one with fewer hops is a spreferred one.

After the Sink node finishes the calculating of the paths, it would send the paths information to node A, where we assumed that the communication between node A and Sink node is trusted. The XML language is used to express the path information. It is extremely simple, compatible with the existing agreements, unified management data access format, sharing and interaction of data between different applications etc., all of which makes XML ideal for data transmission.

- routing establishment

Node A receives the path information transmitted from Sink node, and then to construct the routing information reaching node B.

Here the value of M is critical. The higher the M is, the more successful the detection is. But the following is the increasing energy consumption of nodes. When selecting the value of M, it is needy to make a tradeoff between detection rate and power consumption.

The algorithm mentioned above is to establish M disjoint shortest paths between credible nodes and suspicious nodes, which are used to detect whether nodes are attacked in the network.

2.2. Detection process

Trusted node forwards the routing request packets on M paths established. And we confirm whether the suspicious nodes are malicious or not by replied messages by suspicious nodes described in Fig 2.

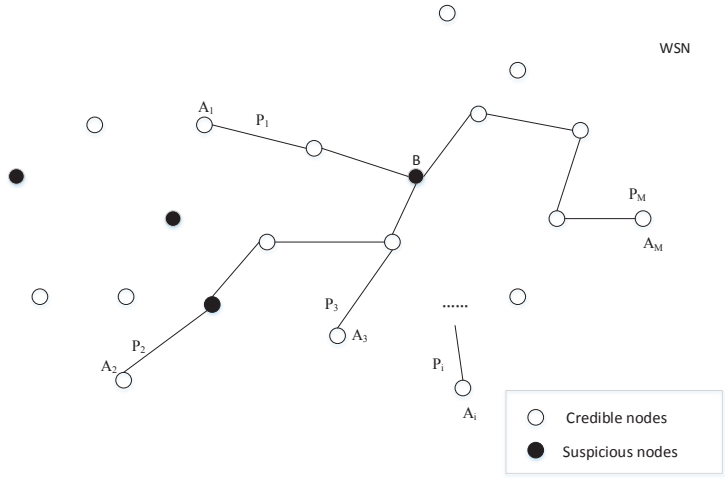

Fig. 2. Multi-path detection

After node A_i receives a reply from node B, it would determine whether the next hop address in the replied packet is same to node B. If it's the same, the replied packet would be not processed; if not, the information of the suspicious node B is sent to the Sink node. Sink node would add up the corresponding information to determine whether the node is malicious or not and the table structure used to justify is as follows:

Table 2. Decision table

S_{sus}	P_1	P_2	P_3	·····	P_M	flag
1	-	-	-	-	-	-
2	-	-	-	-	-	-
·····	.....	.....	.....	.....	.....	.....
Nsus	-	-	-	-	-	-

In Table 2, $\mathbf{N_{sus}}$ represents the number of set $\mathbf{S_{sus}}$ and flag represents whether the node is malicious or not, whose value is true or false. After all the suspicious nodes shake hands with credible nodes, fill the blank of the Table 2 with their own identity. The next is to do a summarization of the number of cell whose content is equal

to its X-axis, which is named as $NumOfB_i$. If the proportion of $NumOfB_i$ becomes more than α, then we determine that B_i is a malicious node as shown in the formula (4). The value of α can be customized and has a close relation of the accuracy of detection of Sinkhole attacks are closely related.

$$\frac{NumOfB_i}{M} \geq \alpha \tag{4}$$

Meanwhile, for cells whose content is not equal to its X-axis, add the content of the cell to set $\mathbf{S_x}$. For each element named as x_i in set $\mathbf{S_x}$ and the node exists in $\mathbf{S_{sus}}$, N_{xi} is the number x_i appear in $\mathbf{S_x}$. We use the formula (5) to determine if the node x_i is malicious or not. N_{sus} represents the number of the nodes in set $\mathbf{S_{sus}}$.

$$\frac{N_x}{M * N_{sus}} \geq \beta \tag{5}$$

The realization of the Sinkhole detection algorithm is as follows:

Step 1. Set the threshold num1 = 0, num2 = 0.

Step 2. If num1≥N_{sus} then jump to **Step 6**; otherwise, num1 = num1 +1, multi-path handshake with the node B_{num1}.

Step 3. If num2≥M, then jump to **Step 2**; otherwise, num2 = num2 +1, credible node A_{num2} send a routing request packet to suspicious node B_{num1} through multiple paths Pi (i = 1, 2, 3 ....., M) .

Step 4. Node A_{num2} receives routing reply packet from B_{num1} and extract the nexthop field.

Step 5. Compare the values of the next hop and B_{num1}.If they are the same, do nothing; otherwise, return the next hop value to the Sink node. And the Sink node counts the corresponding values. Jump to **Step 2**.

Step 6. According to the information received, Sink node determine whether the suspicious nodes are malicious.

Step 7. Set the threshold num1 = 0 again.

Step 8. If num1≥N_{sus}, then jump to **Step 10**; otherwise, num1 = num1 +1, and count $NumOfB_i$. Meanwhile, add the node except B_i to the set $\mathbf{S_x}$.

Step 9. If $\frac{NumOfB_i}{M} \geq \alpha$ ， then the node Bi is added to the set S_{mal}. Jump to **Step 8**.

Step 10. Calculate $\mathbf{S_x}=\mathbf{S_x}-\mathbf{S_{sus}}\cap\mathbf{S_x}$.

Step 11. Set the threshold num1 = 0 again.

Step 12. If num1 ≥ Nx, it finishes; otherwise, num1 = num1 +1, count N_x.

Step 13 If $\frac{NumOfX_i}{M*N_{abnormal}} \geq \beta$， then the node Xi add to the set S_{mal}. Jump to **Step 12**.

3. Experimental results

3.1. Experiment setting

The simulations are performed to prove the feasibility of Sinkhole detection. The simulation tool, NS2, is used do some experiments. In the simulations, the nodes are static and randomly distributed. The scope of activities is 1000m * 1000 meters. The network topology is shown in Fig 3, where red nodes is the attacked nodes by Sinkhole and black ones are normal. At this time, there are five attacked nodes numbered 1-5 and we set node 0 as the Sink node.

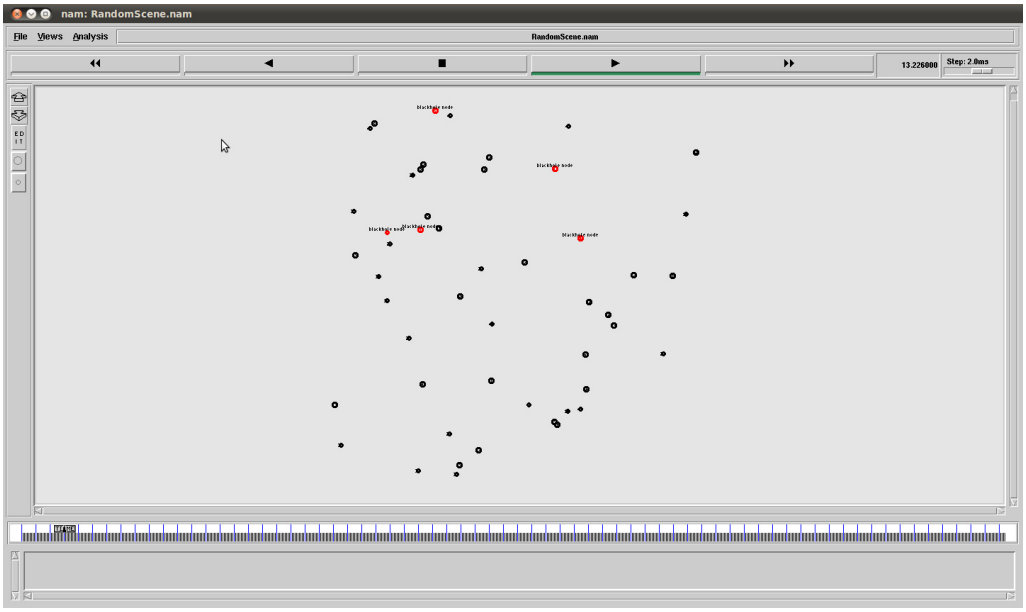

Fig. 3. Network topology

3.2. Experimental results and analysis

In simulation experiments, the detection rate, the mistake rate and miss rate of the Sinkhole attack are used to verify the detection algorithm proposed in this paper.

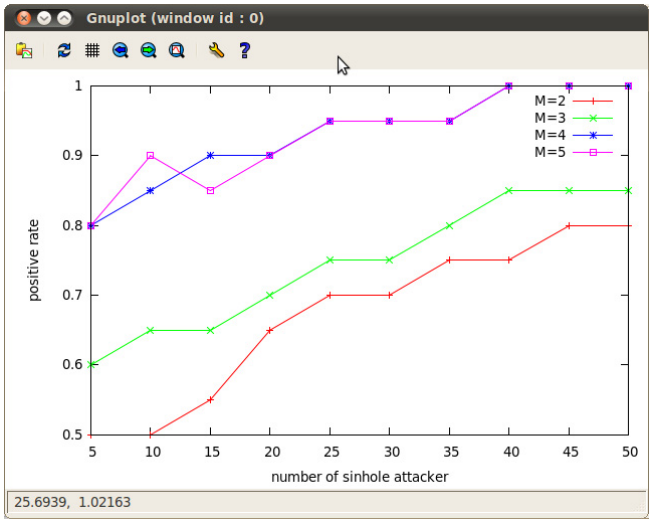

Fig. 4. the comparison of detection rates based on multi-path detection

Fig 4 depicts the comparison of detection rates based on multi-path detection. It's clearly seen that when M is 4 or more, the detection rates of the attack are similar. Considering the efficiency of the algorithm and the energy consumption of nodes, M is set to 4.

In addition, some experiments are done by adding detection algorithm to common routing protocols AODV and DSR protocols in the wireless sensor networks shown in Fig 5. As we can see from the figure that, the detection rate of the DSR protocol is generally more efficient than the AODV protocol with the number of attacked nodes increasing.

The detection algorithm raised in the paper is compared with the classical detection algorithm- BM in Fig 6. From the figure, it concludes that the algorithm here has a higher detection rate. And the Sinkhole detection algorithm proposed could effectively detect the nodes attacked in wireless sensor networks.

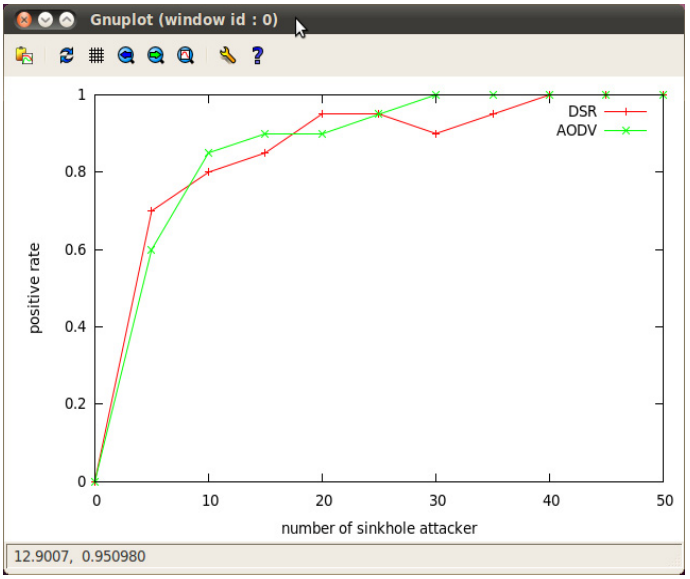

Fig. 5. chart of detection rate comparison

There are many causes of mistake detection and the Sinkhole detection rate of 100% is not realistic. we can only try to high the detection rate. In addition, CBR data streams are transmitted in the same frequency in the simulation, thus data congestion tends to conflict resulting in some errors of the detection algorithm.

Derived from the experiments, Sinkhole attack greatly affects the performance of the network. And with the number of attacked nodes increasing in the network, it is more destructive. It's of important significance to study detection algorithm of Sinkhole attack.

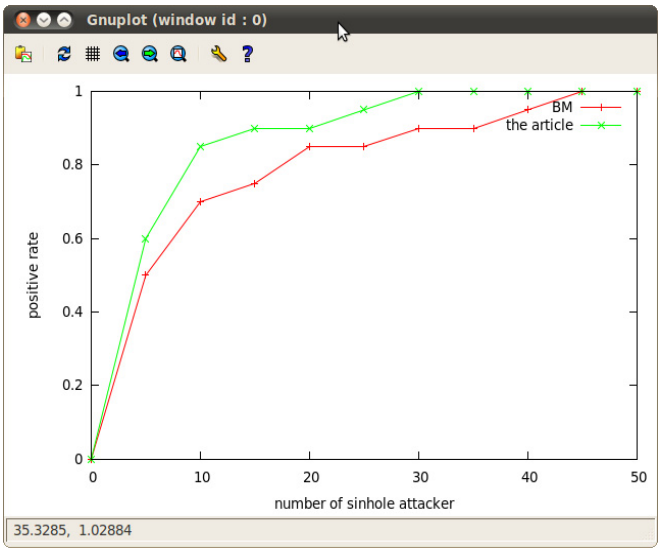

Fig. 6. The comparison of detection rates using different detection algorithm

4. Conclusion

In the paper, the main contribution is to propose a new Sinkhole detection algorithm based the multi-path selection. The simulation also proves the feasibility of the approach. In our future work, we will perfect the algorithm raised continuously and extensively, where there are still many problems existing. Meanwhile, we will improve the simulation more visually. Beyond that, the algorithm would be practiced.

Acknowledgements

This paper is supported by 863 Program (Grant No.2011AA01A103).

References

[1] S. Yi, P. Naldurg, R. Kravets, "A Security-Aware Routing Protocol for Wireless Ad Hoc Networks," the 6th World Multi-conference on Systemic Cybernetics and Informatics 2002:286-292.
[2] Y. C. Hu, D. B. Johnson, A. Perrig, "SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks," the 4th IEEE Workshop on Mobile Computing and Applications 2002:3-13.
[3] M. Al-shurmanm, S. M. Yoo, S. Park, "Black hole attack in mobile Ad hoc networks," the 42nd ACM Southeast Regional Conference New York ACM Press, 2003, 96-97.
[4] S. Mart, T. J. Giuli, Kevin Lai, M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," International Conference on Mobile Computing and Networking New York ACM Press, 2000, 255-265.
[5] I. Aad, J. P. Hubaux, E. W. Knightly, "Denial of Service Resilience in Ad Hoc Networks," the 10nd Annual International Conference on Mobile Computing and Networking, 2004, 202-215.
[6] J. R. Lundberg, "Security in Ad hoc networks[EB/OL]," 2000, http://citeseer.nj.nec.com/400961.html.

[7] B. Awerbuch, D. Holmer, C. N. Rotaru, H. Rubens, "An On-Demand Secure Routing Protocol Resilient to Byzantine Failures," the 1st ACM Workshop on Wireless Security, New York: ACM Press, 2002: 21-30.
[8] Z. N. Peng, D. X. Ye, M. Y. Fan, "Research on black hole attack in mobile Ad hoc networks," Application Research of Computers, vol. 26, No. 11, November,2009.
[9] Yu Qun, "Research on black hole attack in mobile Ad hoc networks," Master Thesis, Jiangsu University, 2006.
[10] Xu Jinhong, "Detection and Location of Malicious Nodes in Wireless Sensor Networks," Master Thesis, Central South University, 2009.